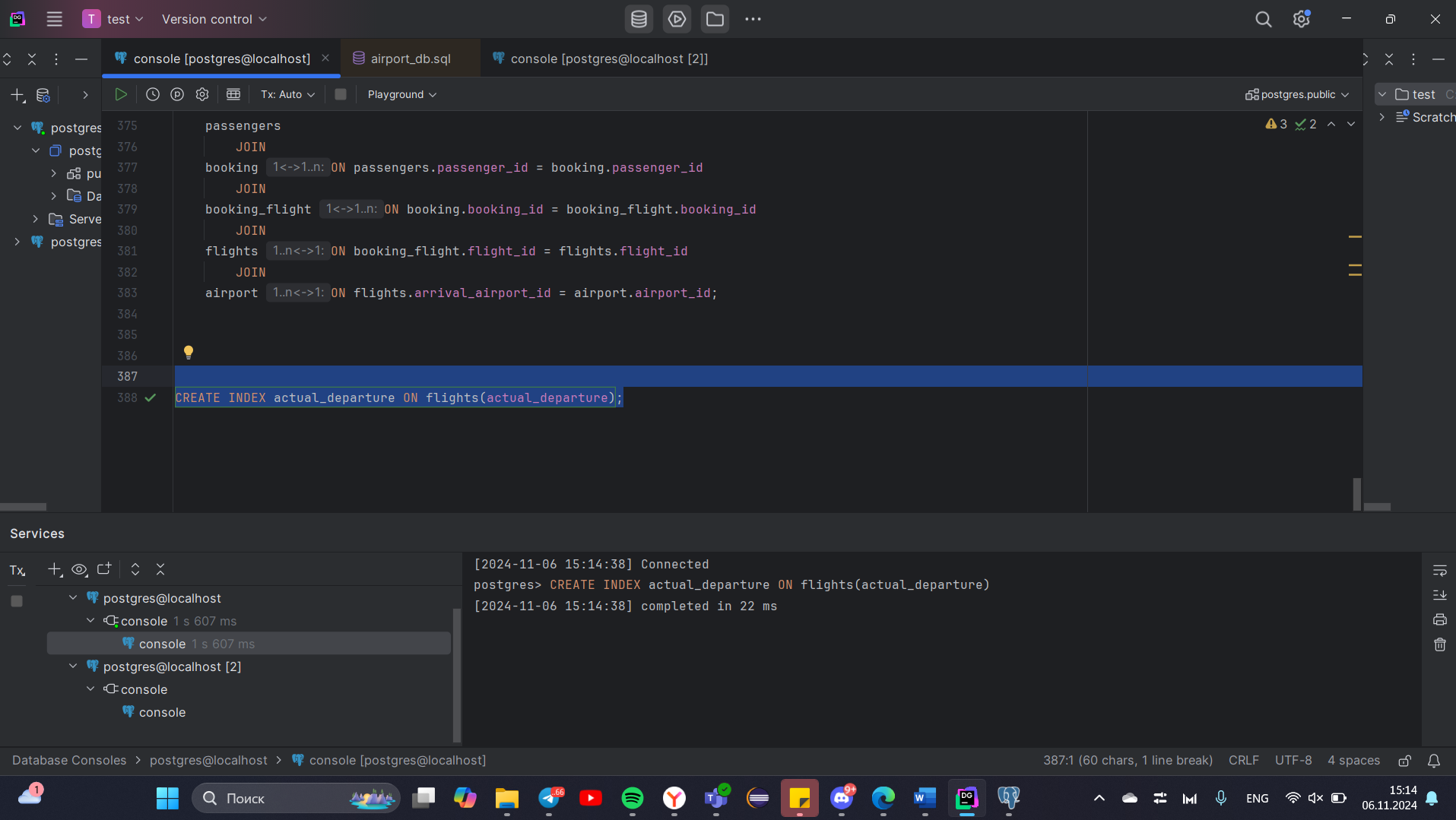
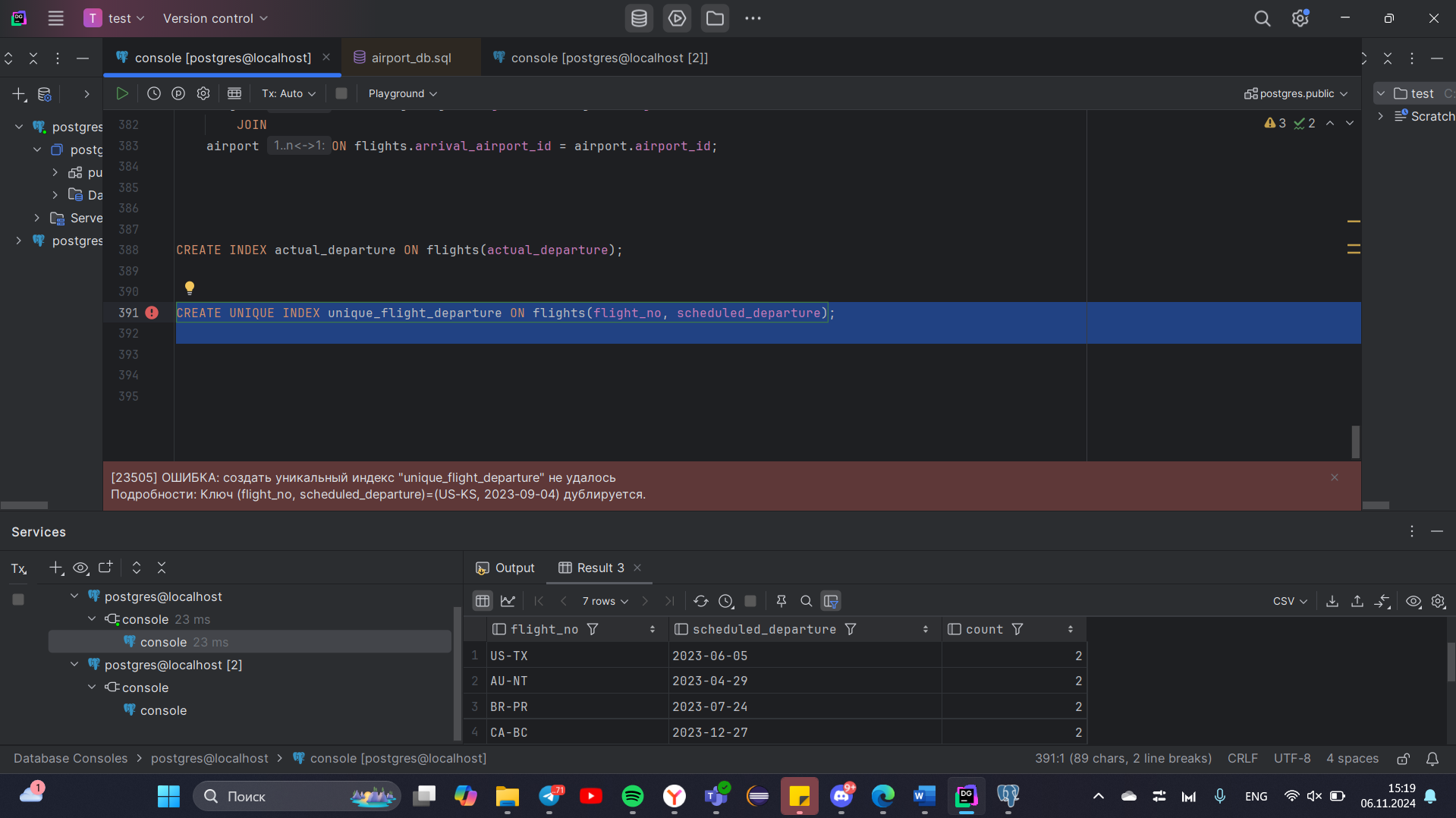
# Laboratory work 7

Tasks:

1. Create an index on the actual\_departure column in the flights table.

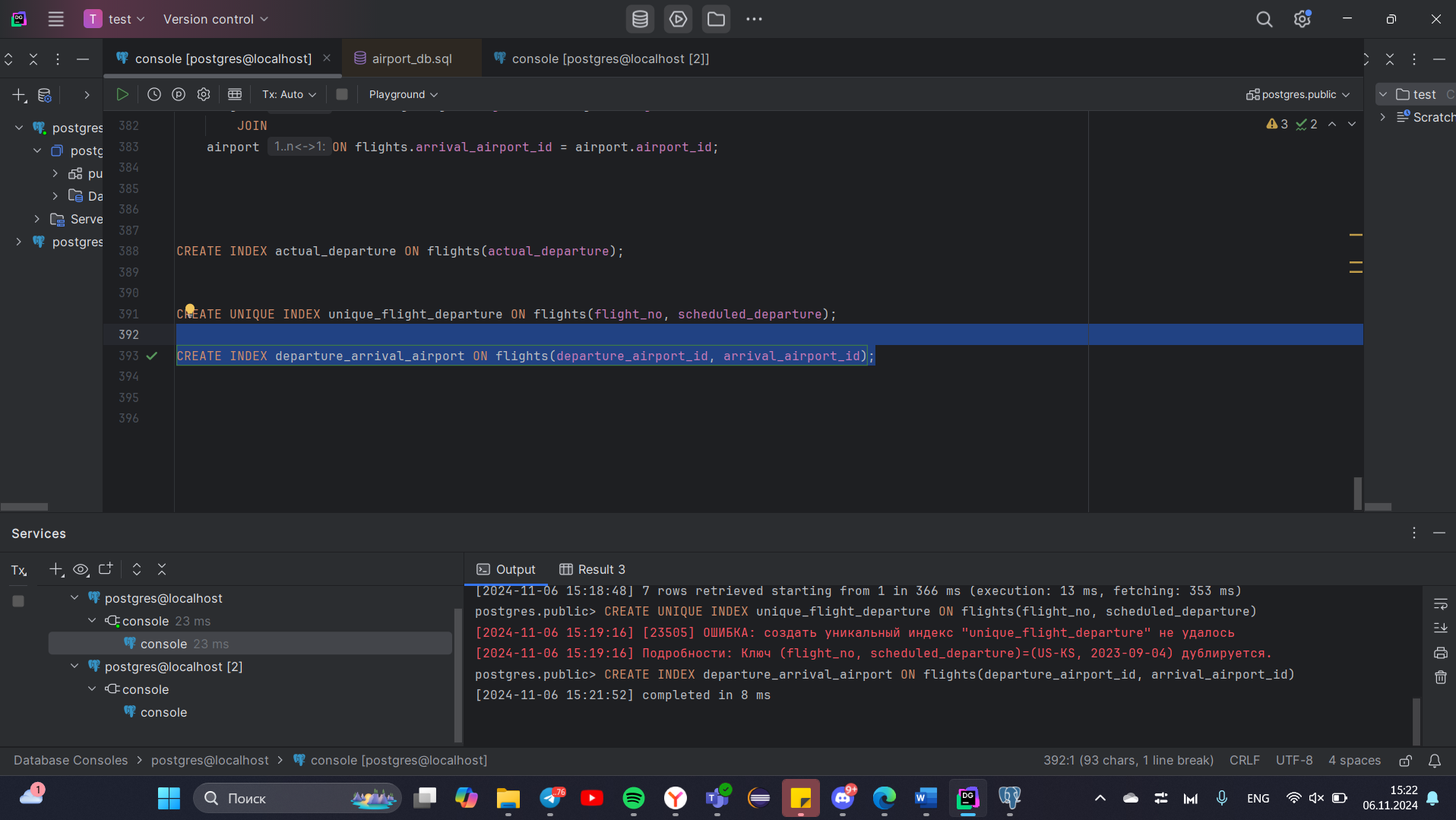


1. Create a unique index to ensure flight\_no and scheduled\_departure combinations are unique.

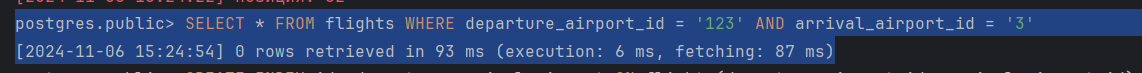


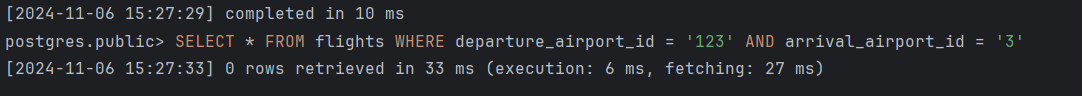
We have dublicates in this colomn so our UNIQUE index can’t be createn

1. Create a composite index on the departure\_airport\_id and arrival\_airport\_id columns.



1. Evaluate the difference in query performance with and without indexes. Measure performance differences.





In output we see that performance with index is faster

1. Use EXPLAIN ANALYZE to check index usage in a query filtering by departure\_airport and arrival\_airport.

Before:

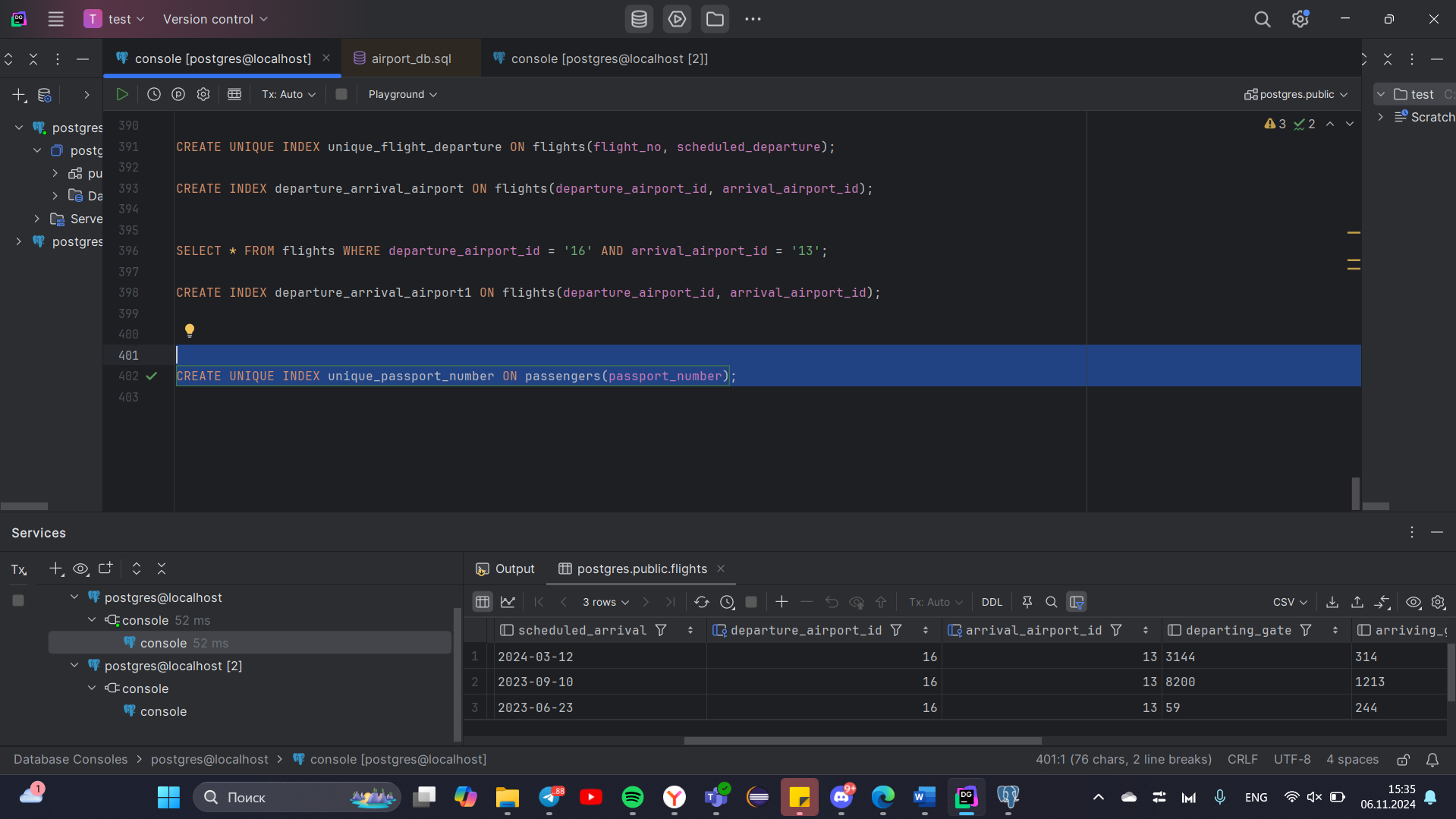
Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

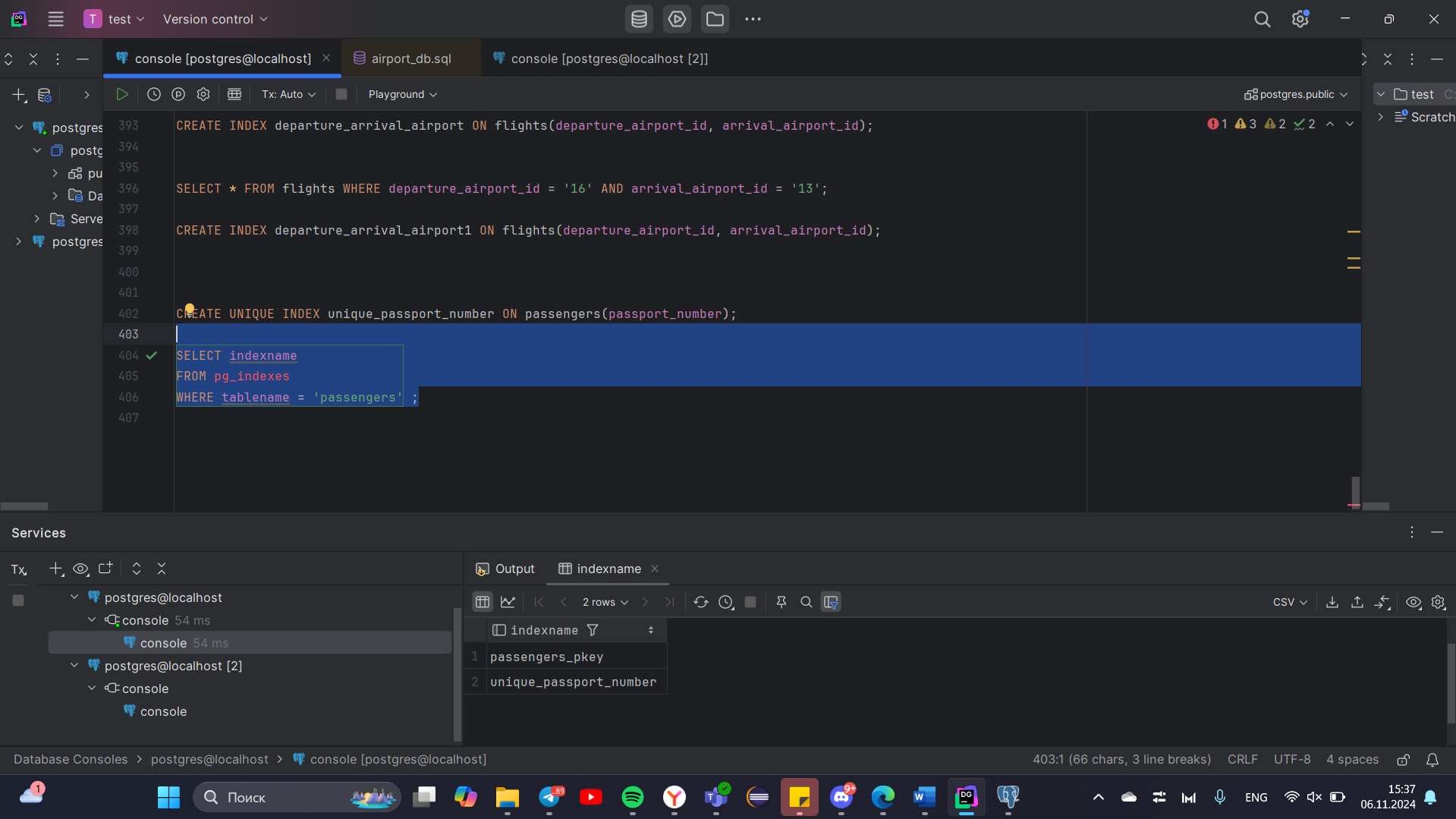
Автоматически созданное описание

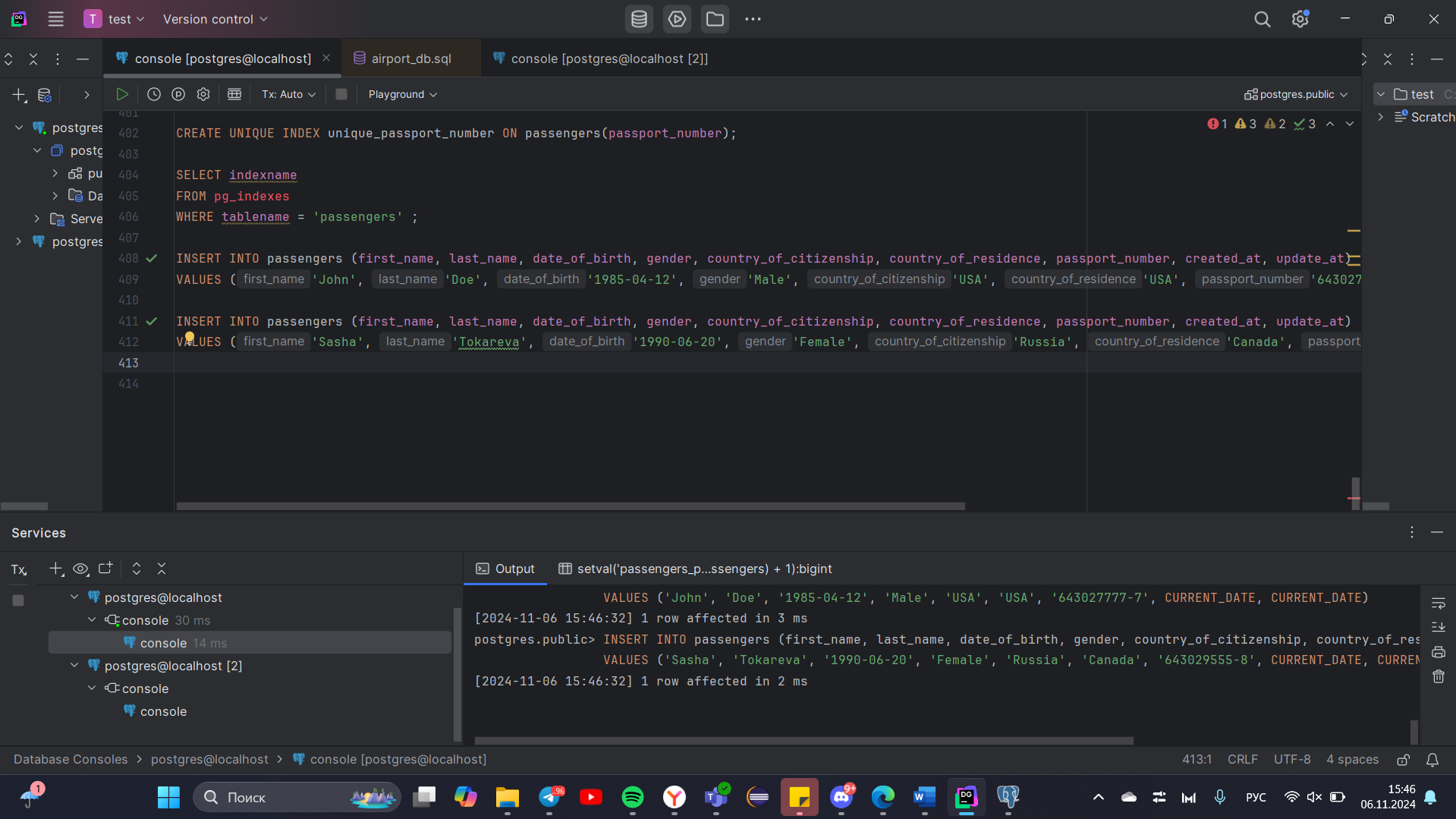
After creating: Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Автоматически созданное описание

1. Create a unique index for the passport\_number of the Passengers table. Check if the index was created or not. Insert into the table two new passengers.

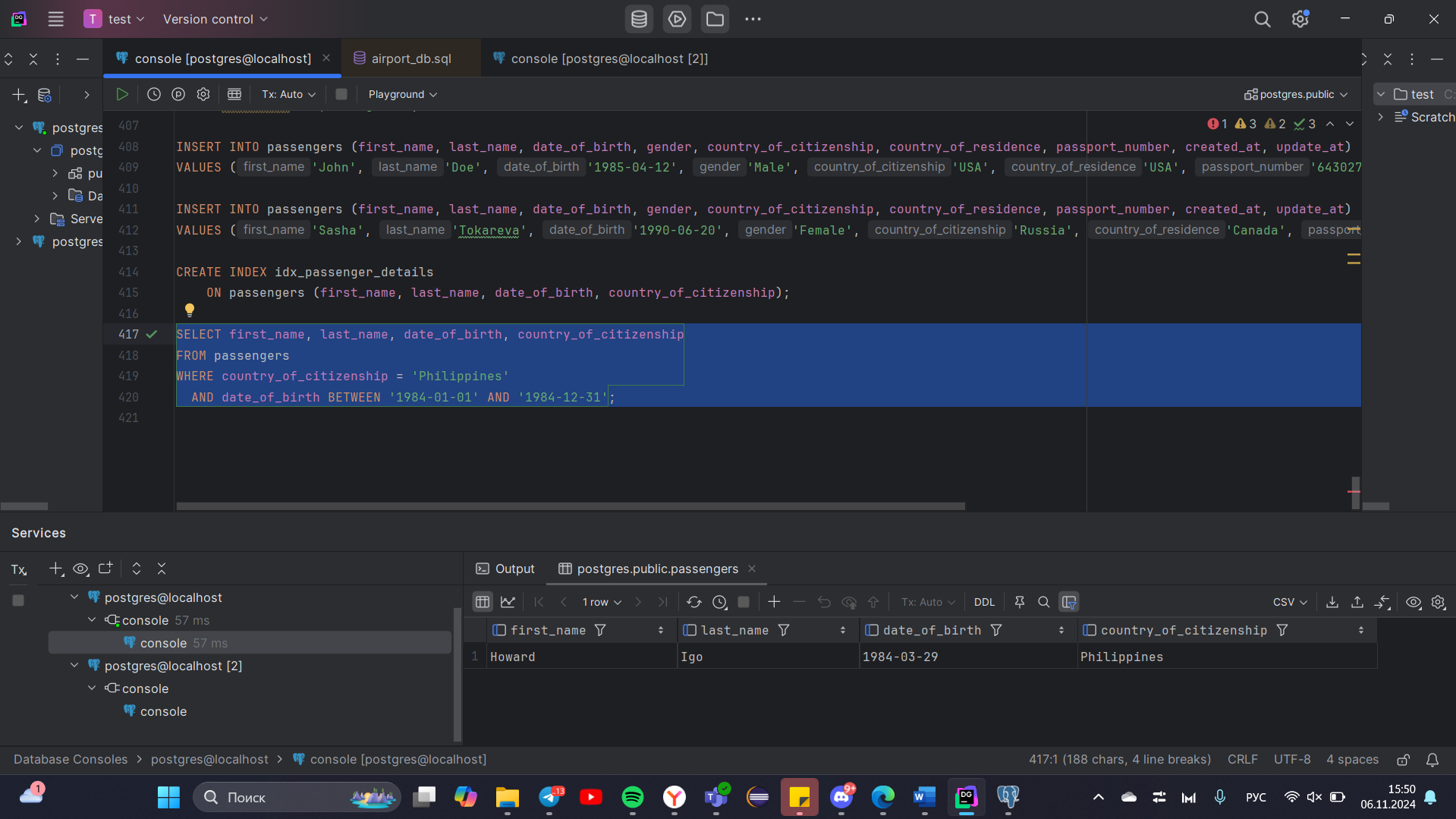


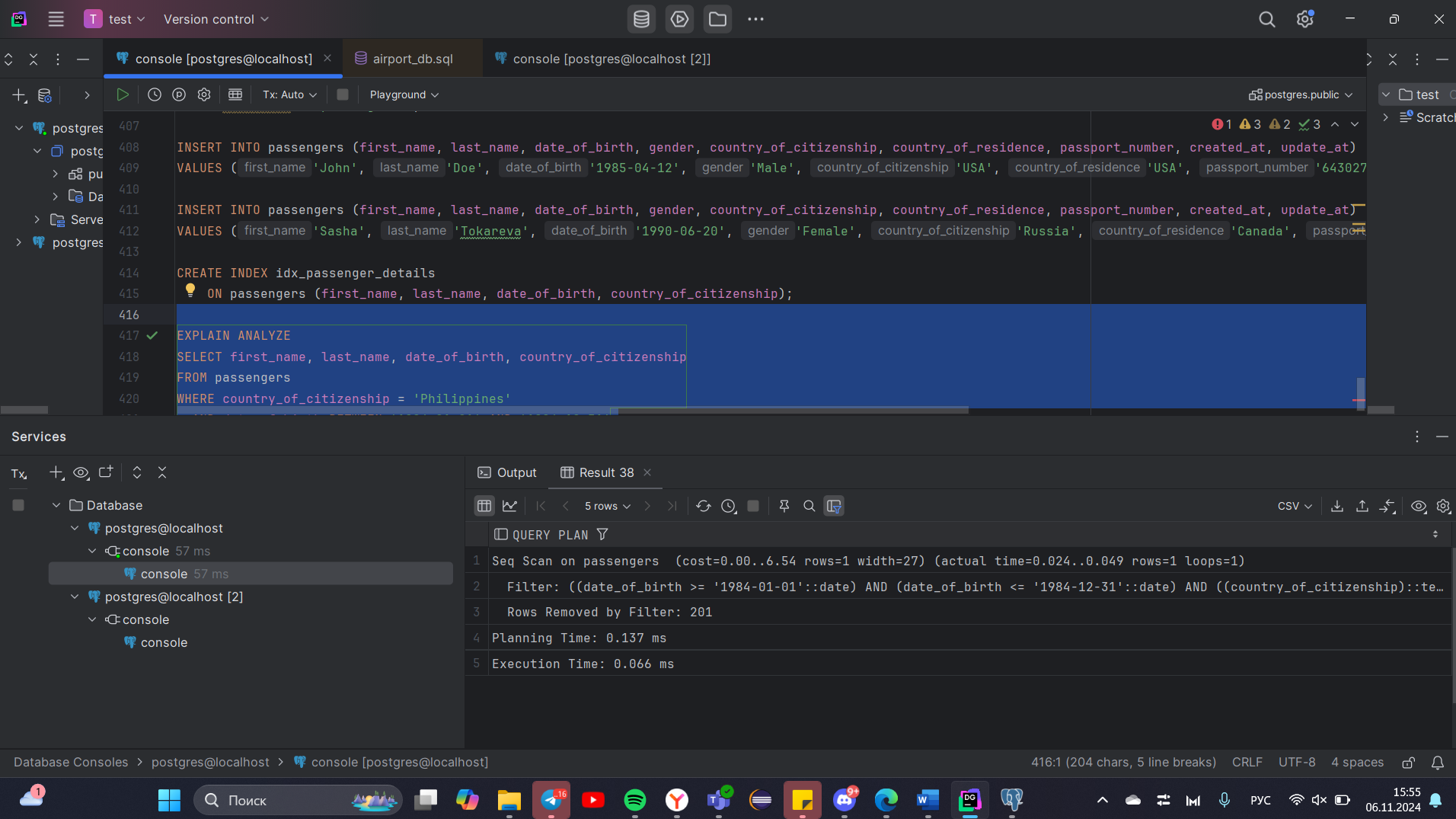




Explain in your own words what is going on in the output?

1. Create an index for the Passengers table. Use for that first name, last name, date of birth and country of citizenship. Then, write a SQL query to find a passenger who was born in Philippines and was born in 1984 and check if the query uses indexes or not. Give the explanation of the results.





Simple finding by query is faster then by index

8. Write a SQL query to list indexes for table Passengers. After delete the created indexes.

